

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 06-348793
(43)Date of publication of application : 22.12.1994

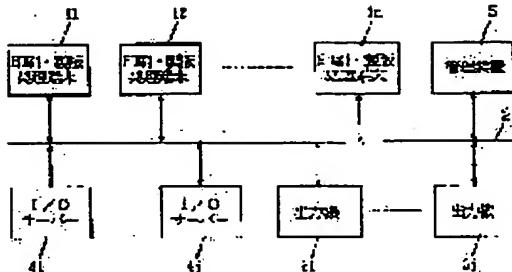
(51)Int.Cl. G06F 15/62
G03F 1/00
G06F 3/12

(21)Application number : 05-160230 (71)Applicant : DAINIPPON SCREEN MFG CO LTD
(22)Date of filing : 03.06.1993 (72)Inventor : NAKANISHI HIDETOSHI

(54) IMAGE PROCESSING SYSTEM

(57)Abstract:

PURPOSE: To provide an image processing system with satisfactory operation efficiency for each output device.
CONSTITUTION: When the output request of image data is issued from each of print/plate making processors 11-1n, a managing device 5 checks the working status of respective output machines 31-3j, and assigns the output request to the output machine set in an inoperative state when such output machine exists, and assigns the output request to the output machine which completes output processing earliest when all the output machines are in operable states. I/O servers 41-4j standly until the output processing of the output machine to which the output request is issued and the image data is assigned is completed.



LEGAL STATUS

[Date of request for examination] 02.12.1996
[Date of sending the examiner's decision of rejection]
[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]
[Date of final disposal for application]
[Patent number] 2992426
[Date of registration] 15.10.1999
[Number of appeal against examiner's decision of rejection]
[Date of requesting appeal against examiner's decision of rejection]
[Date of extinction of right]

THIS PAGE BLANK (USPTO)

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平6-348793

(43) 公開日 平成6年(1994)12月22日

(51) Int.Cl.⁵
G 0 6 F. 15/62
G 0 3 F. 1/00
G 0 6 F. 3/12

識別記号 庁内整理番号
A 8125-5L
L
D

F I

技術表示箇所

審査請求 未請求 請求項の数3 FD (全11頁)

(21) 出願番号 特願平5-160230

(22) 出願日 平成5年(1993)6月3日

(71) 出願人 000207551

大日本スクリーン製造株式会社
京都府京都市上京区堀川通寺之内上る4丁
目天神北町1番地の1

(72) 発明者 中西 英俊

京都市上京区堀川通寺之内上る4丁目天神
北町1番地の1 大日本スクリーン製造株
式会社内

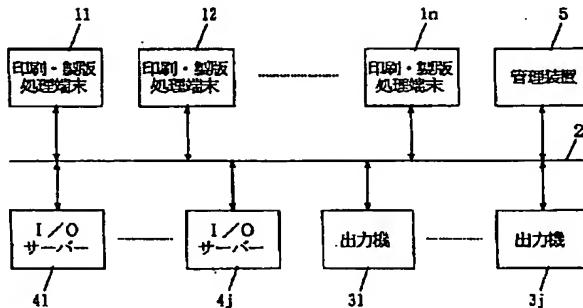
(74) 代理人 弁理士 小笠原 史朗

(54) 【発明の名称】 画像処理システム

(57) 【要約】

【目的】 各出力装置の稼働効率が良好な画像処理システムを提供することである。

【構成】 各印刷・製版処理装置 11～1n から画像データの出力要求が発行されると、管理装置 5 は各出力機 31～3j の稼働状況を調べ、非稼働状態の出力機が存在する場合は当該非稼働状態の出力機に出力要求を割り当てる。全ての出力機が稼働状態の場合は最も早く出力処理を終了する出力機に出力要求を割り当てる。各 I/O サーバー 41～4j は、出力要求の発行された画像データを割り当てられた出力機の出力処理が終了するまで待機させる。



【特許請求の範囲】

【請求項1】 画像データを加工・編集して所定の媒体に出力する画像処理システムであって、
与えられた画像データを加工・編集する複数の画像処理端末、

前記各画像処理端末とネットワーク結合され、前記各画像処理端末によって加工・編集された画像データを所定の媒体へ出力する複数の出力装置、および前記各画像処理端末の動作を管理する管理装置を備え、

前記管理装置は、前記画像処理端末からの画像データの出力要求を、いずれかの出力装置が非稼働状態の場合は当該非稼働状態の出力装置に割り当て、全ての出力装置が稼働状態の場合は最先に出力動作を終了する出力装置に割り当てる出力要求割当手段を含む、画像処理システム。

【請求項2】 前記画像処理端末によって加工・編集された画像データを一時的に記憶保持する記憶手段をさらに備え、

前記出力要求割当手段は、現在稼働中の出力装置に対して前記画像データの出力要求を割り当てるとき、当該出力装置がすでに割り当て済みの出力処理を終了するまで、前記画像処理端末からの画像データを前記記憶手段で待機させる、請求項1に記載の画像処理システム。

【請求項3】 前記各画像処理端末は、前記画像データの出力要求とともに、当該画像データの容量に関連する属性情報を前記管理装置に発行する手段を含み、

前記出力要求割当手段は、前記属性情報に基づいて、最先に出力動作を終了する出力装置を算出する、請求項1または2に記載の画像処理システム。

【発明の詳細な説明】**【0001】**

【産業上の利用分野】 この発明は、画像処理システムに関し、より特定的には、印刷・製版処理システムのように画像データを加工・編集し、所定の媒体に出力するシステムに関する。

【0002】

【従来の技術】 近年、従来の大型汎用コンピュータに代わって、処理能力が著しく向上しているパーソナルコンピュータやワークステーション等の小型コンピュータを活用してLAN(ローカル・エリア・ネットワーク)等を組む動きが活発化している。こうした小型機への移行は、「ダウンサイジング」と呼ばれ、主としてコストの削減や開発期間の短縮化を目的としている。印刷、製版等の画像処理の分野においても、ダウンサイジングが急速に進められている。例えば、分散配置された複数台の画像処理端末がネットワーク環境下で動作し、並行的に所定の画像処理を実行するような画像処理システムが一般的になりつつある。

【0003】 図14は、従来の印刷・製版処理システムの一例を示すブロック図である。図14において、複数

台の印刷・製版処理端末11, 12, …1nは、有線または無線の通信路2を介して複数台の出力機31, …3jとネットワーク結合されている。印刷・製版処理端末11, 12, …1nは、それぞれパーソナルコンピュータやワークステーション等によって構成され、通信路2を介して与えられる印刷・製版データを加工・編集する。出力機31, …3jは、それぞれプリンタやイメージセッタ等によって構成され、印刷・製版処理端末11, 12, …1nによって加工・編集された印刷・製版データをフィルム、刷版、紙等の記録媒体に出力する。

【0004】 図14の印刷・製版処理システムでは、印刷・製版処理端末11, 12, …1nから出力機31, …3jへの印刷・製版データの出力は、以下のようにして行われる。まず、オペレータが印刷・製版処理端末側から適当な1つの出力機を指定し、その出力機に対して出力要求を出す。このとき、指定された出力機が非稼働状態(空き状態)であれば、当該出力機は出力要求を出した印刷・製版処理端末に対してデータ転送命令を発行する。もし、当該出力機が稼働状態(出力処理実行中)であれば、当該出力機は出力要求を出した印刷・製版処理端末を出力待機状態にしておく。出力要求は、それが発行された順にキューとして管理される。

【0005】 図15は、従来の印刷・製版処理システムの他の例を示すブロック図である。図15の印刷・製版処理システムは、1/Oサーバー4を用いている点が図14の印刷・製版処理システムと異なっている。図15の印刷・製版処理システムでは、印刷・製版処理端末11, 12, …1nからの印刷・製版データは、一旦1/Oサーバー4にプール(一時保管)される。印刷・製版処理端末11, 12, …1nは、出力機31, …3jの状態(出力処理実行中、出力処理待ちキュー有り/無し)に関係なく出力処理(データ出力)を完了することができる。ただし、図14の印刷・製版処理システムと同様に、オペレータは印刷・製版処理端末側から、出力機の指定を行わなければならない。

【0006】

【発明が解決しようとする課題】 上記のように従来の印刷・製版処理システムでは、印刷・製版処理端末側でオペレータが出力処理を実行する出力機を指定する必要があった。そのため、オペレータが各出力機の稼働状況を考慮せずに出力機を指定すれば、一部の出力機に出力処理が集中する結果となり、システム全体としての稼働効率が低下するという問題点があった。この問題点を、図16を参照して、より具体的に説明する。なお、以下の説明においてJOBとは、例えば受注情報に基づいて管理された仕事の単位を言い、印刷・製版データの集合体で構成される。例えば出力機31, 32がそれぞれJOB1, JOB2の出力処理を実行中にJOB3の出力要求を発行する場合、印刷・製版処理端末のオペレータは、出力機31, 32がいずれも稼働状態であるため、

とりあえず出力機31に出力要求を発行したとする。この場合、図16に示すように、JOB2の出力処理がJOB1の出力処理よりも早く終了するにもかかわらず、JOB3の出力処理はJOB1の出力処理が終了する時刻t3まで開始されない。その結果、出力機32に無駄な非稼働期間(t3-t2)が発生し、出力機の稼働効率が低下する。なお、印刷・製版処理システム等の画像処理システムで扱われる画像データは、データ量が大きいので、出力処理に長時間を要し、稼働効率の低下は深刻な問題となる。

【0007】それゆえに、この発明の目的は、常に効率よく出力処理が実行できる画像処理システムを提供することである。

【0008】

【課題を解決するための手段】請求項1に係る発明は、画像データを加工・編集して所定の媒体に出力する画像処理システムであって、与えられた画像データを加工・編集する複数の画像処理端末、各画像処理端末とネットワーク結合され、各画像処理端末によって加工・編集された画像データを所定の媒体へ出力する複数の出力装置、および各画像処理端末の動作を管理する管理装置を備え、管理装置は、画像処理端末からの画像データの出力要求を、いずれかの出力装置が非稼働状態の場合は当該非稼働状態の出力装置に割り当て、全ての出力装置が稼働状態の場合は最先に出力動作を終了する出力装置に割り当てる出力要求割当手段を含む。

【0009】請求項2に係る発明は、請求項1の発明において、画像処理端末によって加工・編集された画像データを一時的に記憶保持する記憶手段をさらに備え、出力要求割当手段は、現在稼働中の出力装置に対して画像データの出力要求を割り当てたとき、当該出力装置がすでに割り当て済みの出力処理を終了するまで、画像処理端末からの画像データを記憶手段で待機させる。

【0010】請求項3に係る発明は、請求項1または2の発明において、各画像処理端末は、画像データの出力要求とともに、当該画像データの容量に関連する属性情報を管理装置に発行する手段を含み、出力要求割当手段は、属性情報に基づいて、最先に出力動作を終了する出力装置を算出する。

【0011】

【作用】請求項1に係る発明においては、画像処理端末から画像データの出力要求が発行されたとき、管理装置の出力要求割当手段は、各出力装置が最も効率良く稼働するように、当該出力要求を出力装置へ割り当てる。すなわち、いずれかの出力装置が非稼働状態の場合は当該非稼働状態の出力装置に出力要求を割り当て、全ての出力装置が稼働状態の場合は最先に出力動作を終了する出力装置に出力要求を割り当てる。

【0012】請求項2に係る発明においては、出力要求割当手段は、現在稼働中の出力装置に対して画像データ

の出力要求を割り当てたとき、画像処理端末からの画像データを記憶手段に一時的に格納する。これによって、当該出力装置がすでに割り当て済みの出力処理を終了するまで、当該出力要求の実行が待機させられる。

【0013】請求項3に係る発明においては、各画像処理端末は、画像データの出力要求とともに、当該画像データの容量に関連する属性情報を管理装置に発行する。そして、出力要求割当手段は、上記属性情報に基づいて、最先に出力動作を終了する出力装置を算出する。

【0014】

【実施例】図1は、この発明の一実施例の印刷・製版処理システムの構成を示すブロック図である。図1に示すように、本実施例の印刷・製版処理システムは、有線または無線の通信路2を介して、複数台の印刷・製版処理端末11, 12, …1nと、複数台の出力機31, …3jと、各出力機31, …3jに1対1に対応する複数台のI/Oサーバー41, …4jと、1台の管理装置5とが、データ通信可能にネットワーク結合された構成となっている。

【0015】図2は、図1に示す各印刷・製版処理端末11, 12, …1nの構成の一例を示すブロック図である。図2において、システムバス100には、CPU101, メインメモリ102, フレームバッファ103, 磁気ディスク装置105, キーボード107およびマウス108が接続される。中央処理装置としてのCPU101は、所定の印刷・製版処理を実行する。メインメモリ102は、CPU101が印刷・製版処理を実行するときに作業用メモリとして用いられる。フレームバッファ103は、表示用メモリであり、CRTディスプレイ110に写し出す1フレーム分の画像データを記憶する。フレームバッファ103に記憶された画像データは、D/A変換器109でアナログ映像信号に変換された後、CRTディスプレイ110に与えられ、そこに表示される。

【0016】磁気ディスク装置105には、各種プログラムデータ(OSシステム、ウインドウシステム、アプリケーションプログラム等)および印刷・製版データが格納されている。CPU101は、磁気ディスク装置105に格納されたプログラムデータに従って動作し、磁気ディスク装置105に格納された印刷・製版データに対して加工・編集処理を実行する。入力装置としてのキーボード107およびマウス108は、オペレータによって操作され、CPU101にコマンドや各種データを入力する。さらにシステムバス100には、通信路2との間に通信インターフェイス111が接続される。

【0017】図3は、図1に示す各I/Oサーバー41, …4jの構成の一例を示すブロック図である。図3に示すように、I/Oサーバーは、図2の印刷・製版処理端末と同様に、システムバス400, CPU401, メインメモリ402, フレームバッファ403, 磁気デ

ィスク装置405、キーボード407、マウス408、D/A変換器409、CRTディスプレイ410および通信インタフェイス411を備えている。

【0018】図4は、図1に示す管理装置5の構成の一例を示すブロック図である。図4に示すように、管理装置5は、図2の印刷・製版処理端末または図3のI/Oサーバーと同様に、システムバス500、CPU501、メインメモリ502、フレームバッファ503、磁気ディスク装置505、キーボード507、マウス508、D/A変換器509、CRTディスプレイ510および通信インタフェイス511を備えている。

【0019】図5は、各印刷・製版処理端末11, 12, …1nにおけるオペレータの操作手順を示すフローチャートである。図6は、各印刷・製版処理端末11, 12, …1nの動作を示すフローチャートである。図7は、管理装置5の動作を示すフローチャートである。図8は、各I/Oサーバー41, …4jの動作を示すフローチャートである。図9は、各出力機31, …3jの動作を示すフローチャートである。以下、これら図5～図9を参照して、図1に示す印刷・製版処理システムにおける各種操作ないし動作の中から、本発明にとって関係のある部分の操作ないし動作を説明する。

【0020】まず、図5を参照して、各印刷・製版処理端末11, 12, …1nにおいてJOBデータの出力要求を発行する際の操作手順を説明する。今、CRTディスプレイ110には、図10(a)に示すように、アイコンIC1～IC8が表示されているものとする。アイコンIC1～IC6は、それぞれ、対応するJOB1～JOB6のファイルを示している。アイコンIC7は、出力機を示している。アイコンIC8は、データ削除時に用いられるごみ箱を示している。出力要求の発行操作は、いわゆるドラッグ&ドロップと呼ばれる方法によって行われる。すなわち、オペレータは、マウス108を操作して、カーソルを出力処理の対象となるJOBアイコン(図10(a)では、JOBアイコンIC3)の上に移動させる(ステップS11)。カーソルがJOBアイコンIC3の上に位置すると、オペレータは、マウス108に付属するボタン(マウスボタン)を押圧する(ステップS12)。次に、オペレータは、マウスボタンを押圧しながら出力機アイコンIC7の上にカーソルを移動させる(ステップS13)。このとき、JOBアイコンIC3は、カーソルの移動に伴って移動する。カーソルが出力機アイコンIC7の上に位置すると、すなわち図10(b)に示すように、JOBアイコンIC3が出力機アイコンIC7の上に重なると、オペレータはマウスボタンの押圧を解除する(ステップS14)。これで、CPU101は、JOB3のデータファイルに対して出力要求が出されたことを認識する。

【0021】上記のごとく、本実施例では、各印刷・製版処理端末11, 12, …1nのオペレータは、各出力

機31, …3jの稼働状況を調べることなく、また出力処理を実行する出力機を指定することなく出力要求を発行できる。すなわち、本実施例では、以下に説明するように各出力機31, …3jへの出力要求の割り当ては、全て管理装置5が管理してくれる。したがって、印刷・製版処理端末のオペレータの操作が単純で簡単なものとなる。

【0022】図5の操作が実行されると、対応する印刷・製版処理端末のCPU101は、図6に示す動作を実行する。まず、図11に示すようなJOB情報(出力要求の対象となるJOBについての各種属性を示す情報)を作成し、通信インタフェイス111を介して当該JOB情報を管理装置5に送出する(ステップS21)。図11に示すJOB情報は、一例として、仕事を受注したクライアントの名前と、受注No.と、受注担当者の名前と、納期と、JOB名(例えば、印刷物の1単位に対応している)と、JOBのID(識別番号)と、印刷物の色数と、ページのサイズと、ページ数と、総データ量などを含んでいる。管理装置5は、印刷・製版処理端末から受け取ったJOB情報に基づいて、出力処理のスケジューリングを実行する。

【0023】ここで、図7を参照して、管理装置5の動作を説明する。管理装置5のCPU501は、まず出力機31, …3jのいずれかから出力処理終了メッセージが送られてきたか否かを判断する(ステップS31)。後述するように、各出力機31, …3jは、1つのJOBデータの出力処理が終了すると、管理装置5に出力処理終了メッセージを送出する(図9のステップS53)。出力処理終了メッセージを受信すると、CPU501は、図12に示すような出力処理スケジュールテーブル中の対応するJOBの項目に出力終了時間を書き込む(ステップS32)。この出力処理スケジュールテーブルは、メインメモリ502内に格納されており、各印刷・製版処理端末11, 12, …1nからJOB情報が送られてくる毎に対応するJOBの項目が追加される。各項目は、一例として、JOB名と、JOBのIDと、ページサイズと、ページ数と、総データ量と、出力開始予定時間と、現実の出力開始時間と、出力終了予定時間と、現実の出力終了時間と、出力機の番号とを含む。上記ステップS32では、対応するJOBの項目に現実の出力終了時間が書き込まれる。その後、CPU501は、ステップS31の動作に戻る。

【0024】一方、出力機からの出力処理終了メッセージを受信しなかった場合、CPU501は、印刷・製版処理端末11, 12, …1nのいずれかからJOB情報が送られてきたか否かを判断する(ステップS33)。当該JOB情報も受信しなかった場合はステップS31の動作に戻るが、当該JOB情報を受信した場合は受信したJOB情報に基づいて、対応するJOBデータの出力時間(出力機での出力処理に要する時間)を算出する

(ステップS34)。なお、出力時間は、当該JOB情報に含まれる総データ量に基づいて算出される。

【0025】次に、CPU501は、出力処理のスケジューリングを行う(ステップS35)。このステップS35の動作をより詳細に説明すると、CPU501は、まず出力処理スケジュールテーブルを参照して非稼働状態の出力機があるか否かを判断する。なお、出力処理スケジュールテーブル内の各項目には出力機番号が登録されているが、同一の出力機番号に対する項目の全てに出力終了時間が書き込まれている場合は、該当の出力機が非稼働状態であると判断される。また、出力処理スケジュールテーブル内のいずれの項目にも番号が登録されていない出力機も非稼働状態であると判断される。非稼働状態の出力機が有る場合、CPU501は、出力処理スケジュールテーブルに新しいJOBの項目(ステップS31で受信したJOBの項目)を追加するとともに、当該追加したJOBの項目に非稼働状態の出力機の番号を登録する。これによって、非稼働状態の出力機に出力要求が割り当てられる。また、CPU501は、新しく追加したJOBの項目に出力開始予定時間および出力終了予定時間を書き込む。なお、出力開始予定時間としては現在時刻が書き込まれ、出力終了予定時間としては現在時刻にステップS34で算出された出力時間を加算した時間が書き込まれる。

【0026】一方、非稼働状態の出力機が無い場合、すなわち全ての出力機が稼働状態の場合、CPU501は、出力処理スケジュールテーブルに登録された各出力機の最終の出力終了予定時間を比較し、最も早く出力処理を終了するであろう出力機を探し出す。次に、CPU501は、出力処理スケジュールテーブルに新しいJOBの項目を追加するとともに、当該追加したJOBの項目に上記探し出した出力機の番号を登録する。これによって、最も早く出力処理を終了する出力機に出力要求が割り当てられる。また、CPU501は、新しく追加したJOBの項目に出力開始予定時間および出力終了予定時間を書き込む。なお、出力開始予定時間としては上記探し出した出力機の最終の出力予定時間がそのまま書き込まれ、出力終了予定時間としては上記最終の出力予定時間にステップS34で算出された出力時間を加算した時間が書き込まれる。

【0027】次に、CPU501は、JOB情報を発行した印刷・製版処理端末に出力メッセージを送出する(ステップS36)。この出力メッセージには、JOBデータの出力要求が割り当てられた出力機の番号と、当該出力機が非稼働状態か否かを示すフラグ情報を含められる。

【0028】印刷・製版処理端末のCPU101は、管理装置5から上記出力メッセージを受信すると(図6のステップS22)、受信した出力メッセージを解析する(ステップS23)。次に、CPU101は、上記フラ

グ情報に基づいて、出力要求が割り当てられた出力機が非稼働状態か否か判断する(ステップS24)。当該出力機が非稼働状態の場合、CPU101は、当該出力機にJOBデータ(印刷・製版データ)を転送する(ステップS25)。一方、当該出力機が稼働状態の場合、CPU101は、当該出力機に対応するI/OサーバーにJOBデータを転送する(ステップS26)。

【0029】I/OサーバーのCPU401は、印刷・製版処理端末からJOBデータを受信すると(図8のステップS41)、受信したJOBデータを磁気ディスク装置405に格納する(ステップS42)。これによって、当該JOBデータは、出力処理の待機状態となる。また、CPU401は、対応する出力機が現在出力しているJOBデータの出力処理を終了したか否かを観察している(ステップS43)。対応する出力機の出力処理が終了した場合、CPU401は、磁気ディスク装置405で待機中の先頭のJOBデータを対応する出力機に送出する(ステップS44)。

【0030】出力機は、印刷・製版処理端末または対応のI/OサーバーからJOBデータを受信すると(図9のステップS51)、受信したJOBデータをフィルム、刷版、紙等の記録媒体に出力する(ステップS52)。次に、出力機は、出力処理終了メッセージを管理装置5に送出する(ステップS53)。前述したように、管理装置5のCPU501は、出力機から出力処理終了メッセージを受信すると、出力処理スケジュールテーブル中の対応するJOBの項目に出力終了時間を書き込む(図7のステップS32)。なお、上記ステップS52において、出力機は出力処理開始メッセージを管理装置5へ送出する。これを受けて、管理装置5のCPU501は図12の出力処理スケジュールテーブルへ出力開始時間を書き込む。

【0031】上記のように本実施例の印刷・製版処理システムでは、全ての出力機が稼働状態の場合、最終の出力終了予定時間が最も早い出力機に出力処理が割り当てられる。その結果、図13に示すように、出力機31よりも出力機32の方が早く出力処理が終了する場合、出力機32にJOB3の出力処理が割り当てられる。したがって、図16の場合と異なり、出力機32に無駄な非稼働期間が生じることなく、システム全体の稼働効率が向上する。

【0032】なお、上記実施例では、出力機で出力処理が終了したJOBの項目を出力処理スケジュールテーブルから消さずに残すようにしている。これは、出力処理スケジュールテーブルの格納情報を後に(例えば、1日の終了後に)管理情報として利用するためである。その後、出力処理が終了したJOBの項目が出力処理スケジュールテーブルから削除される。出力処理スケジュールテーブルの格納情報を管理情報として利用しない場合は、出力機で出力処理が終了する毎に(例えば、図7の

ステップS32で)出力処理スケジュールテーブルから対応するJOBの項目を削除するようにしてもよい。この場合、出力処理スケジュールテーブルを格納するメインメモリ502の容量を節約できる。

【0033】また、上記実施例では、出力予定時間をデータ量に基づいて算出したが、その他の要素例えはページサイズ等の出力サイズに基づいて出力予定時間を算出してもよい。

【0034】また、上記実施例では、出力機31～3jに1対1に対応するようにI/Oサーバー41～4jを設けるようにしたが、1つのI/Oサーバーで複数の出力機を受け持つようにしてもよい。

【0035】また、上記実施例では、I/Oサーバーを設けてJOBデータ(画像データ)を一時保管させたが、この一時保管手段はI/Oサーバーに限られるものではない。

【0036】また、上記実施例は、印刷・製版処理システムとして構成されたが、この発明はネットワーク環境下で運用されるその他の画像処理システム(例えば、半導体のマスクパターン作成装置、コンピュータグラフィック作成装置、建築設計図面作成装置)にも広く適用が可能である。

【0037】

【発明の効果】請求項1に係る発明によれば、画像処理端末から画像データの出力要求が発行されたとき、いずれかの出力装置が非稼働状態の場合は当該非稼働状態の出力装置に出力要求を割り当て、全ての出力装置が稼働状態の場合は最先に出力動作を終了する出力装置に出力要求を割り当てるようになっているので、各出力装置を効率良く稼働させることができ、結果としてシステム全体の生産性が向上する。

【0038】請求項2に係る発明によれば、稼働中の出力装置に対して画像データの出力要求を割り当たときは、画像データを記憶手段において待機させるようしているので、出力要求を発行した画像処理端末が画像データの出力処理から素早く切り離される。その結果、当該画像処理端末は他の処理を実行可能となり、システム全体の生産性が低下することがない。

【0039】請求項3に係る発明によれば、画像データの容量に関連する属性情報に基づいて最先に出力動作を

終了する出力装置を算出するようになっているので、正確な算出が行える。

【図面の簡単な説明】

【図1】この発明の一実施例の印刷・製版処理システムの全体構成を示すブロック図である。

【図2】各印刷・製版処理端末の詳細な構成を示すブロック図である。

【図3】各I/Oサーバーの詳細な構成を示すブロック図である。

【図4】管理装置の詳細な構成を示すブロック図である。

【図5】各印刷・製版処理端末におけるオペレータの操作手順を示すフローチャートである。

【図6】各印刷・製版処理端末の動作を示すフローチャートである。

【図7】管理装置の動作を示すフローチャートである。

【図8】各I/Oサーバーの動作を示すフローチャートである。

【図9】各出力機の動作を示すフローチャートである。

【図10】出力要求発行時における印刷・製版処理端末のCRTディスプレイの一表示例を示す図である。

【図11】印刷・製版処理端末で発行される属性情報の一例を示す図である。

【図12】管理装置のメインメモリに格納される出力処理スケジュールテーブルの一例を示す図である。

【図13】図1の印刷・製版処理システムの利点を説明するための図解図である。

【図14】従来の印刷・製版処理システムの一例を示す図である。

【図15】従来の印刷・製版処理システムの他の例を示す図である。

【図16】従来の印刷・製版処理システムの出力処理動作を示す図である。

【符号の説明】

1 1～1n：印刷・製版処理端末

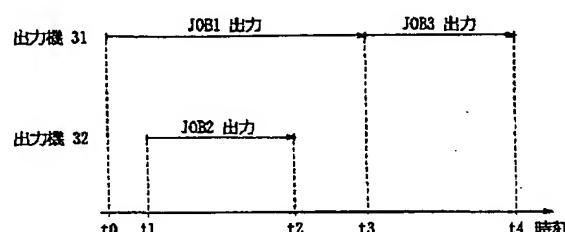
2：通信路

3 1～3j：出力機

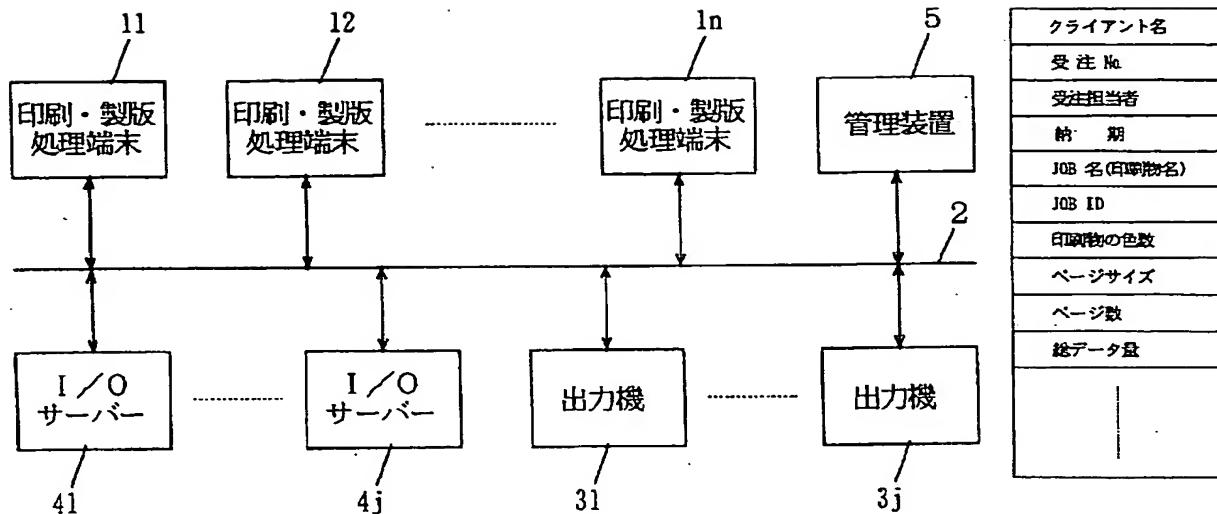
4 1～4j：I/Oサーバー

5：管理装置

【図16】

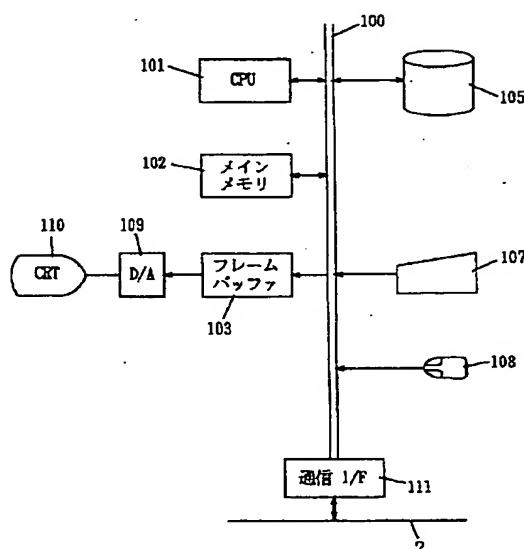


【図1】

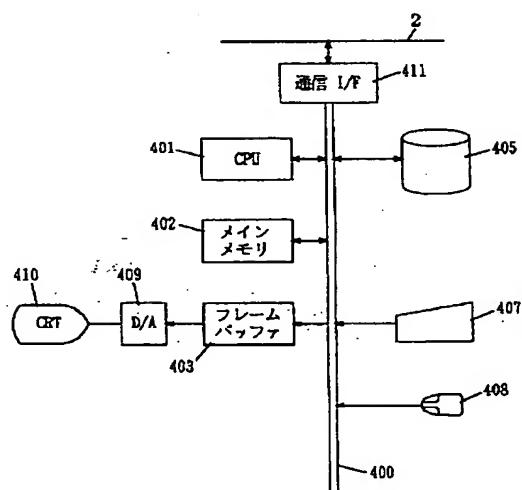


【図11】

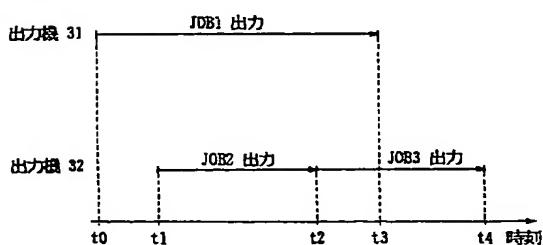
【図2】



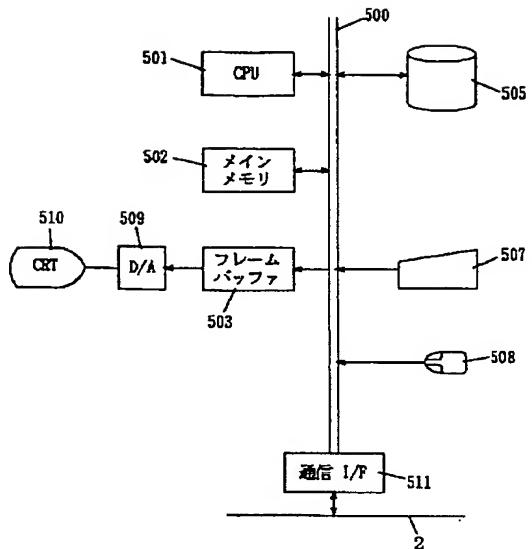
【図3】



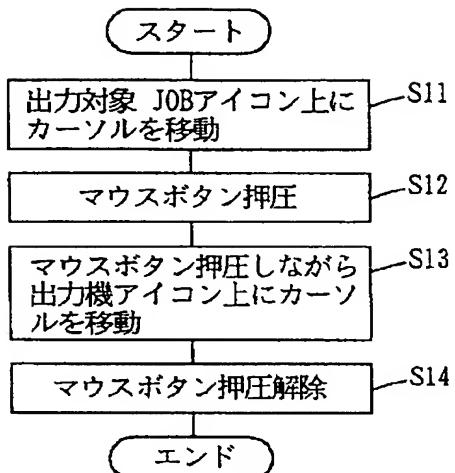
【図13】



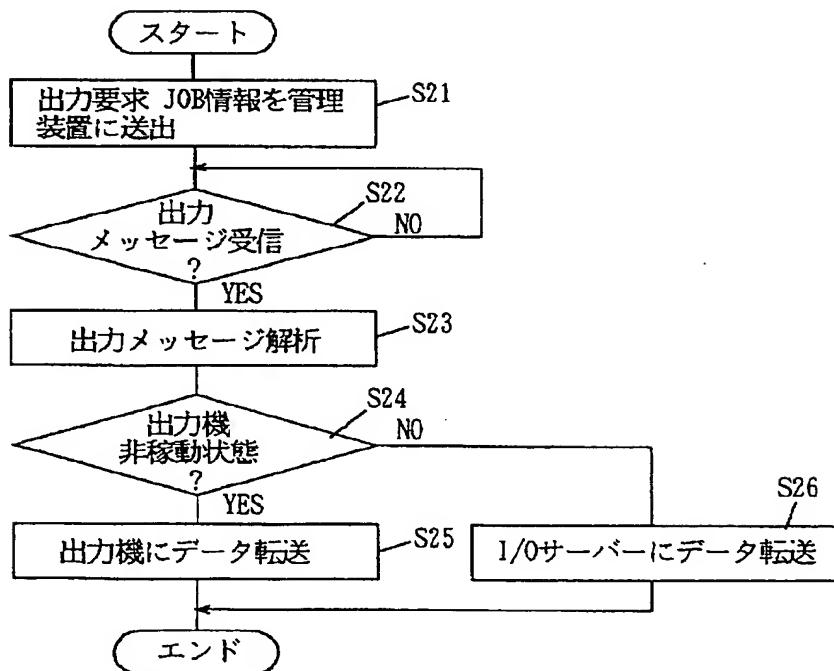
【図4】



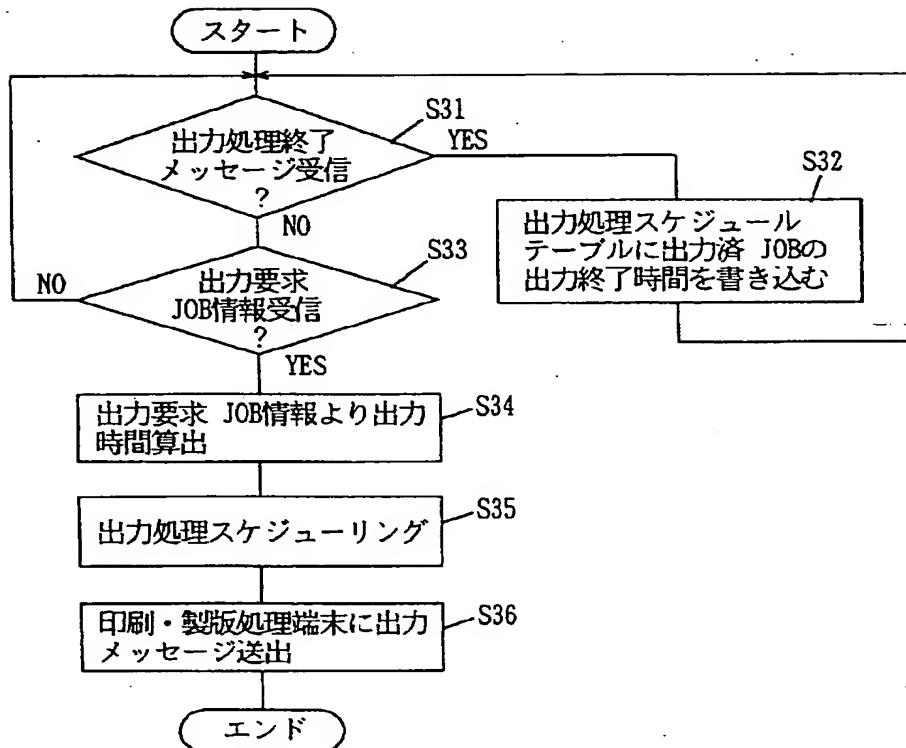
【図5】



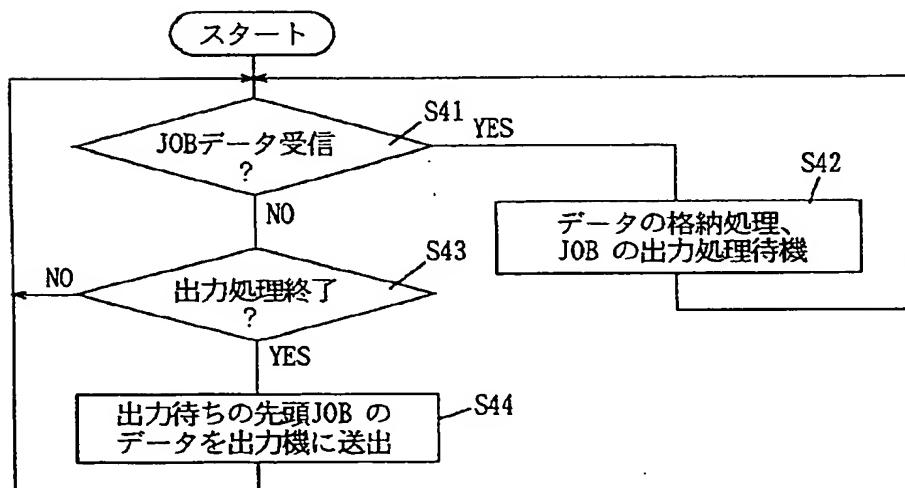
【図6】



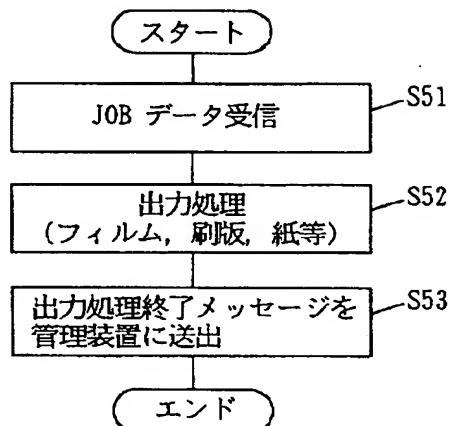
【図7】



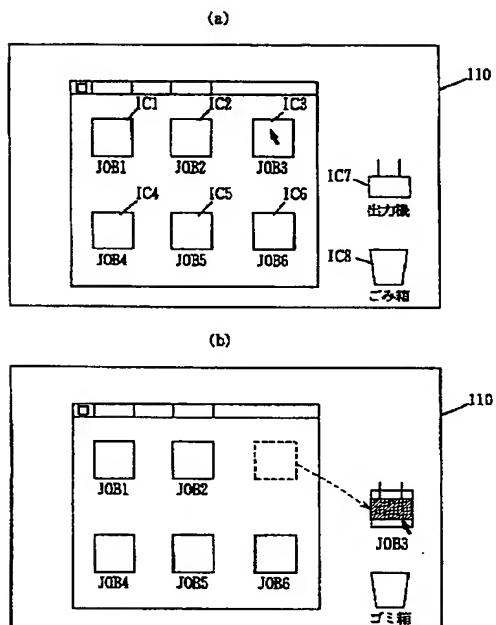
【図8】



【図9】



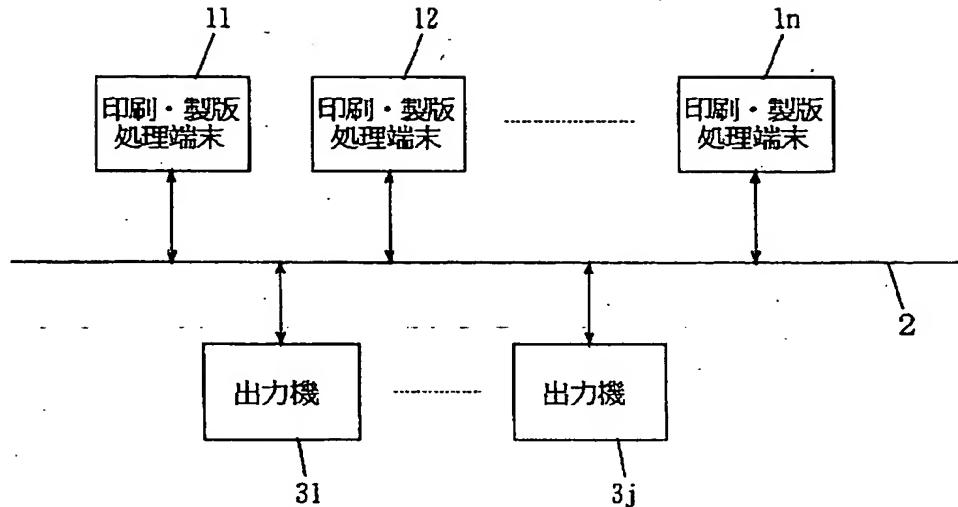
【図10】



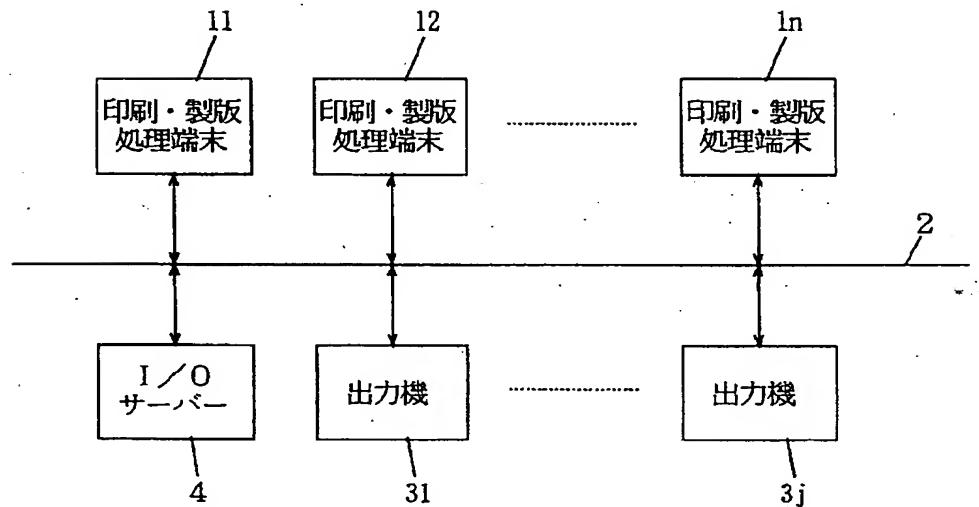
【図12】

Job名	Job ID	頁サイズ	頁数	給データ量	出力開始予定期間	出力開始時間	出力終了予定期間	出力終了時間	出力機番号
screen1	10000	A4	10	600MB	9:00.00	9:00.00	9:30.00		1
screen2	10001	A4	20	1.2GB	9:00.00	9:00.00	10:00.00		2
screen3	10002	A4	20	1.2GB	9:30.00		10:30.00		1
photo1	11001	A3	10	1.2GB	10:00.00		11:00.00		2
photo2	11002	A3	20	2.4GB	10:30.00		12:30.00		1
photo3	11003	A3	10	1.2GB	11:00.00		12:00.00		2
ad1	12001	A4	20	1.2GB	12:00.00		13:00.00		2
ad2	12002	A4	30	3.6GB	12:30.00		14:00.00		1

【図14】



【図15】



THIS PAGE BLANK (USPTO)